METHOD OF OFFERING FREE PRODUCTS OR SERVICES OVER THE INTERNET

CROSS-REFERENCE TO RELATED APPLICATIONS

This application claims the benefit of the filing date of U.S. Provisional Application No. 60/204,801, filed on May 16, 2000.

BACKGROUND OF THE INVENTION

FIELD OF THE INVENTION

This invention relates to commerce conducted over the Internet and particularly to the offer of products or services for sale over the Internet.

DESCRIPTION OF THE RELATED ART

The purchase and sale of goods and services over the Internet by consumers has been growing at a rapid rate. Although there are many advantages to consumers in conducting such electronic commerce transactions, such as the low cost and speed of the transactions, the ease of employment of the Internet, the virtually unlimited range of products and services offered, the security of payments, and the anonymity of transactions, many consumers are still hesitant to purchase over the Internet due to privacy or security concerns, the novelty of the methods involved, or for many other unknown or not easily articulated reasons.

The subject invention offers an additional incentive to hesitant consumers to purchase over the Internet, besides the other advantages previously cited. The invention provides this incentive by allowing a consumer of goods or services to acquire those goods or services at

no cost according to a random process, and the consumer can be notified, prior to an order, of the probability that he or she will receive the goods or services desired to be purchased at no cost.

Thus, for example, a consumer may be informed that if he wishes to purchase a particular book, every tenth book ordered will be delivered at no charge. Alternatively, the consumer may be informed that the number of books ordered before a free one is offered will be a random number. In addition, a consumer may be notified of the number of purchase orders placed for a particular good or service.

SUMMARY OF THE INVENTION

The invention comprises a method of offering free product(s) and/or service(s) over the Internet embodied by, for example, a computer software program for E-Commerce applications (hereinafter sometimes referred to as "**LUCK(Y)CYCLE**"). The program is an enhancement to existing merchant web-sites which would enable the merchant to offer free product(s) and/or service(s) to customers in accordance with pre-set parameters chosen by the merchant as part of his marketing strategy.

The software program comprises four user-definable algorithms allowing the merchant user to predict the probability of a free product and/or service being offered to the customer.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a flow chart of an Internet purchase transaction without the use of the subject invention.

Fig. 2 is a flow chart of an internet purchase transaction utilizing the subject invention.

Figs. 3 and 3A are source code for the default **LUCK(Y)CYCLE** management screen available for each individual and/or (a) group(s) of product(s) and/or service(s) in an online store's catalog.

Figs. 4, 4A, and 4B are source code for result screens for **LUCK(Y)CYCLE** showing the winning product(s) and/or service(s), depending on the merchant's choice of one of the four user-definable algorithms.

Figs. 5, 5A, and 5B are source code for the four **LUCK(Y)CYCLE** user-definable algorithms.

Fig. 6 shows a **LUCK(Y)CYCLE** management screen where the merchant has selected the regular cycle algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) in the on-line store's catalog.

Fig. 7 shows a **LUCK(Y)CYCLE** result screen showing the individual and/or (a) group(s) of product(s) and/or service(s) offered for free corresponding to the merchant's choice of the regular cycle algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) as shown in Fig. 6.

Fig. 8 shows a **LUCK(Y)CYCLE** management screen where the merchant has selected the constant probability algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) in the on-line store's catalog.

Fig. 9 shows a **LUCK(Y)CYCLE** result screen showing the individual and/or (a) group(s) of product(s) and/or service(s) offered for free corresponding to the merchant's

choice of the constant probability algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) as shown in Fig. 8.

Fig. 10 shows a **LUCK(Y)CYCLE** management screen where the merchant has selected the pre-defined list algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) in the on-line store's catalog.

Fig. 11 shows a **LUCK(Y)CYCLE** result screen showing the individual and/or (a) group(s) of product(s) and/or service(s) offered for free corresponding to the merchant's choice of the pre-defined list algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) as shown in Fig. 10.

Fig. 12 shows a **LUCK(Y)CYCLE** management screen where the merchant has selected the dynamic probability algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) in the on-line store's catalog.

Fig. 13 shows a **LUCK(Y)CYCLE** result screen showing the individual and/or (a) group(s) of product(s) and/or service(s) offered for free corresponding to the merchant's choice of the dynamic probability algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) as shown in Fig. 12.

DESCRIPTION OF THE PREFERRED EMBODIMENT(S)

E-Commerce and On-line Shopping

E-commerce sites (such as Amazon.com for example) offer customers a large catalog of individual and/or (a) group(s) of products and/or services. As shown in Fig. 1, on

arriving at the on-line shop 1 on the internet using a web browser 2, the customer may browse 4 through the range of products and/or services available for purchase in the on-line catalog 6, progressively select product(s) and/or service(s) for purchase 8, and add them to a virtual "shopping basket" 10 prior to payment 18 (via credit card 16) and exit 20 from the "store" 1.

As shown in Fig. 2, with the addition of the **LUCK(Y)CYCLE** program 22 to an on-line store, the customer could be entitled to benefit from free offers, free products and/or services, etc. 24 in accordance with a strategy pre-defined by the merchant and regulated and managed by the Lucky Cycle program.

The **LUCK(Y)CYCLE** program will enable the merchant to attribute to each individual and/or (a) group(s) of product(s) and/or service(s) in his catalog individualized parameters which will decide the probability of a free gift of that individual and/or (a) group(s) of product(s) and/or service(s) 26 being offered to the customer.

How it Works

General Principle

Each individual and/or group of product(s) and/or service(s) can be characterized by its own algorithmic cycle which will define the statistical probability of it being offered free to the customer. This cycle is pre-defined by the merchant and represents an estimate of the number of items which should be sold in order that one item or group of items may be offered free.

Procedures activated at the moment of purchase of any item or group of items will enable the customer to see whether he receives it or them for free or whether he must pay for it or them.

→ # products avail

In this application, the letter "n" will be used to represent the cycle selected by the merchant. The proposed algorithms will be based, amongst other things, on the number of catalog items of any given type ordered by the total number of customers visiting the site since its opening. Each catalog item ordered will thus have its own index, which is "p", and the cycle selected by the merchant is specific to each catalog item.

↳ winning free article # in cycle n purchases eg 6th article purchased (p) in first 10 purchases (=n) is free,

Example

Assume that an E-commerce site is offering two articles, A and B.

Article A has a cycle where n=10, which means that the probability of it being offered for free is 1/10 or 10%.

Article B has a cycle independent of article A, and which may be different from n=10.

The first article A has an index of p=1.

The second article A has an index of p=2.

The first article B has an index of p=1.

And so on...

Explanation of the Different Algorithms

The regular cycle

In this algorithm, after (n-1) articles have been sold, the nth article is offered for free. The probability is therefore a direct function of "p".

Mathematically, it could be stated that the article is offered for free when (p modulus n) = 0.

This mathematical statement could be extended to a more general equation: (p modulus n) = whole number constant between 0 and (n-1). If we call the whole number constant c, this more general equation would describe the result that after c-1 articles have been sold, the cth article is offered for free for the first cycle, after n+c-1 articles have been sold, the (n+c)th article is offered for free for the second cycle, and so on.

The source code of the regular cycle principle is shown in Fig. 5 underneath the highlighted regular cycle algorithm title.

A screenshot in Fig. 6 shows how the merchant selects this algorithm. The screenshot in Fig. 7 shows the resulting individual and/or group of products and/or services offered for free when n has been set to 10 and p has been set to 50.

↳ 10: every 10 items get 1 free, repeat 10 items etc.   ↳ 50 items purchased

Example

The cycle where n=10 would mean that the 10th, the 20th, the 30th...etc...article ordered would be offered free to the customer.

The constant probability

This algorithm is characterized by an identical probability for all values of "p". For all orders placed for the article, each customer will have a 1/n probability of a free gift.

Mathematically, this cycle is characterized by the generation of a random number between 0 and (n-1). If this number equals 0 (or any other constant between 0 and (n-1)), then the article is offered for free.

mod 0

The source code of the constant probability algorithm is shown in Fig. 5 underneath the highlighted constant probability algorithm title.

A screenshot in Fig. 8 shows how the merchant selects this algorithm. Fig. 9 shows the resulting individual and/or group of products and/or services offered for free when n has been set to 10 and p has been set to 50.

Example

The cycle where n=10 would give all customers for this article a 1 in 10 chance of winning it for free.

The pre-defined list

This algorithm comprises determining at the opening of the site a series of whole numbers included between 1 and v which will determine future winners.

RHS limit

If the index "p" for any particular order corresponds to a number contained within this list, then the article is offered for free.

This list should therefore contain v/n numbers in order to respect the n cycle. When the v articles have been ordered, a new series of numbers must be created between v+1 and 2v.

This series of numbers may be created manually by the site administrator, or at random by a number generator.

Mathematically, v/n distinct numbers are generated with values between 1 and v. If "p" is included in this series, then the article is offered for free.

One particular case in this cycle is where v=n. In this case, the list is comprised of a single element. This "list" is recreated whenever "p" reaches a multiple of n and includes a number to be found between p and (p+n).

The source code of the pre-defined list algorithm is shown in Figs. 5 and 5A underneath the highlighted pre-defined list algorithm title.

A screenshot in Fig. 10 shows how the merchant selects this algorithm. Fig. 11 shows the resulting individual and/or group of products and/or services offered for free when n has been set to 10 and p has been set to 50.

The dynamic probability

This algorithm calculates the probability of obtaining an article for free according to the difference between p and the next article to be found in a pre-defined reference list.

For example, assume a pre-defined reference list of a regular series such as: 10, 20, 30, 40, ... corresponding to a regular cycle where n=10.

At the opening of the site, the next article in the reference list is thus 10. The first article ordered will have a probability of 1 in 10. The second article ordered will have a probability of 1 in 9. The third article ordered will have a probability of 1 in 8. If we assume that this third article is offered for free, then the next available number in the reference list becomes 20. Thus, the fourth article ordered will have a probability of 1 in 17.

ps w/o replacement

next cycle

If p' is the next number in the reference list, then the probability is expressed as 1/(p'-p+1). This algorithm can be generalized by taking any reference list, as long as it always respects the probability of 1/n. The function of probability 1/(p'-p+1) can itself be replaced by any other function of p and p'.

The source code of the dynamic probability algorithm is shown in Fig. 5A underneath the highlighted dynamic probability algorithm title.

A screenshot in Fig. 12 shows how the merchant selects this algorithm. Fig. 13 shows the resulting individual and/or group of products and/or services offered for free when n has been set to 10 and p has been set to 50.

It should be understood that where any of the constant probability, pre-defined list, and dynamic probability algorithms require the generation of a random number, that random number need not be an integer within the desired range of values, but may be a rational fraction as well. The fraction could then be rounded to an integer for further use in the algorithm. The use of such fractional values would have the effect of increasing the possible number of random values generated, but it should not have any effect on the probability of any integer being chosen.